Module Eight Journal: Portfolio Reflection

Ed Morrow

Southern New Hampshire University

CS-405: Secure Coding

Professor Mike Prasad

20th April 2023

Journal: Portfolio Reflection

**Adaption of a secure Coding Standard, and not leaving Security to the End**

Adopting a secure coding standard is a prudent measure to ensure that all developers are on the same page when it comes to security: while many may already be mindful of potential security vulnerabilities and take care to address them as part of their work, others may not; and, with the best will in the world, the level of skill and experience developers can bring to bear will naturally vary considerably. A secure coding standard defines a minimum level of security that would be built in to all aspects of the codebase, and it also facilitates greater transparency and more effective auditing and accounting of security and allows for a coordinated Defense in Depth (DiD) strategy rather than an *ad hoc* patchwork of security measures whose efficacy could be quite inconsistent and whose weak spots would be much harder to track.

Not leaving security to the end goes hand in hand with the adoption of a secure coding standard: the very existence of a standard demonstrates that security was considered from the very beginning of the project and a plan established to work it into every aspect of the code rather than attempting to incorporate it later as an afterthought. This is important because, while no system can be said to be totally secure and new threats are emerging all the time, considering security at every step of the process allows us to make the system as secure as possible; and a firm foundation based on secure coding standards (together with security policies including encryption and triple-A, which we also studied during this course) affords a much better platform to adapt to new, previously unforeseen threats than a system that does not even tackle the basics.

**Evaluation and Assessment of Risk**

The last point discussed above touched on the difficulty inherent in evaluating and assessing risk: since new threats are constantly emerging, the process of implementing security is primarily a reactive one. However, existing, well-known security vulnerabilities can be somewhat objectively evaluated in terms of the potential harm that they might do and the anticipated remediation cost, and it was this criterion which took precedence in the ranking I devised as part of the Green Pace Security Policy: while attempts to assess the likelihood of a security vulnerability being attacked are certainly valuable, they seem to be rather more speculative which led me to assign them a secondary importance in the Policy’s ranking.

Given that resources (time and money, but also RAM and clock cycles!) are always limited, evaluating and assessing risk — however imperfect a science it may be — is always worthwhile because it allows us to target resources where they are most needed to offer the biggest ‘bang for buck’ in terms of security enhancements.

**Zero Trust**

Zero Trust is a security framework that requires that all users be authenticated and authorized with each attempted access and their security clearances continuously and repeatedly validated. This complements triple-A policies by ensuring that they are implemented consistently for all users upon every attempt to access systems.

While Zero Trust may prove slightly irritating for users who have to enter multiple passwords whenever they try to access a system, proper training on security policies should help them to understand that the framework is a valuable part of a DiD strategy and the minor cost of having to pause to input credentials would be more than offset by the security afforded by requiring all users to be authenticated and authorized at all times.

**Implementation and Recommendations of Security Policies**

A security policy can only be as effective as its implementation, so the primary recommendation would be to invest resources in training developers to understand what is required, and to create automation and analysis processes which can support and monitor the implementation of the policies.

Such an investment of resources would require buy-in from stakeholders, so part of the recommendations included when pitching such a policy would need to include the potential benefits of the policy versus the risks of not implementing it: benefits and risks are two sides of the same coin here, since the benefits primarily involve minimizing the risk of financial and reputational damage and even the possibility of systems being taken offline altogether. However, as noted above, there are additional benefits in acting sooner, including potentially being in a position to take the initiative in terms of implementing security rather than constantly attempting to address issues as they arise.

In working on the Security Policy for Green Pace, I feel that I have gained a good overview of what would be necessary to set a reasonable minimum standard for secure coding practices, a topic of which I knew very little prior to this class. In working on the presentation, I also gained some insight into how one might go about persuading executives of the importance of investing in an area which might appear to some to be an optional extra that doesn’t enhance the company’s profits: in this case, focusing on the negative consequences of failure to implement security — bolstered by specific examples such as the Alibaba Data Leak (Hope, 2021), where necessary — might prove an effective stratagem in getting the message across.

References

Hope, A. (2021, June 25). *Web Scraping on Alibaba’s Taobao Resulted in Data Leak of 1.1*

*Billion Records.* CPO Magazine. https://www.cpomagazine.com/cyber-security/web-

scraping-on-alibabas-taobao-resulted-in-data-leak-